1.- DATOS DE LA ASIGNATURA

Nombre de la asignatura :	**Programación en Ambiente Cliente/Servidor**
Carrera :	**Ingeniería Informática**
Clave de la asignatura :	**IFF-1019**
SATCA[1]	**3-2-5**

2.- PRESENTACIÓN

Caracterización de la asignatura.
Esta asignatura aporta al perfil del Ingeniero en Informática la capacidad para analizar, diseñar y desarrollar aplicaciones distribuidas que atiendan y resuelvan las necesidades de información de las organizaciones.

Intención didáctica.
En la primera unidad se analiza el modelo de programación cliente/servidor y se describen las características principales de los modelos de computación distribuida.

De los conocimientos básicos para la creación de una interfaz socket se encarga la unidad dos; aquí los estudiantes identifican las características de los sockets y las utilizan en el desarrollo de aplicaciones cliente/servidor.

En la tercera unidad el estudiante se introduce al conocimiento de modelos computacionales distribuidos analizando el uso de la invocación remota de métodos RMI usando el lenguaje de programación Java, para desarrollar una aplicación cliente/servidor.

El modelo de objetos componentes COM/DCOM es otro modelo computacional distribuido con propósitos semejantes al RMI que está orientado en ambientes propios de Windows.

[1] *Sistema de Asignación y Transferencia de Créditos Académicos*

3.- COMPETENCIAS A DESARROLLAR

Competencias específicas:	Competencias genéricas:	
▪ Conocer los distintos aspectos de programación cliente servidor. ▪ Conocer, diseñar y desarrollar aplicaciones atendiendo la arquitectura cliente servidor. ▪ Utilizar sockets en el desarrollo de aplicaciones cliente/servidor ▪ Seleccionar el modelo de cómputo distribuido pertinente para una aplicación específica. ▪ Crear e implementar un servicio web.	*Competencias instrumentales* • Capacidad de análisis y síntesis. • Capacidad de resolver. • Conocimientos básicos de la carrera. • Comunicación oral y escrita. • Habilidades avanzadas de manejo de la computadora. • Habilidad para buscar y analizar información proveniente de fuentes diversas. • Solución de problemas. • Toma de decisiones. *Competencias interpersonales* • Capacidad crítica y autocrítica. • Trabajo en equipo. • Habilidades interpersonales. *Competencias sistémicas* • Capacidad de aplicar los conocimientos en la práctica. • Habilidades de investigación. • Capacidad de aprender. • Capacidad de generar nuevas ideas (creatividad). • Capacidad para diseñar y gestionar proyectos. • Habilidad para trabajar en forma autónoma. • Búsqueda del logro.	

4.- HISTORIA DEL PROGRAMA

Lugar y fecha de elaboración o revisión	Participantes	Evento
Instituto Tecnológico de Saltillo del 5 al 9 de octubre de 2009.	Representantes de los Institutos Tecnológicos de: Apizaco, Cerro Azul, Chetumal, Ciudad Juárez, Ciudad Madero, Superior de Coatzacoalcos, Colima, Comitancillo, Conkal, Durango, El Llano Aguascalientes, El Salto, Superior de Fresnillo, Huejutla, Superior de Lerdo, Linares, Los Mochis, Mexicali, Morelia, Oaxaca, Superior del Occidente del Estado de Hidalgo, Ocotlán, Orizaba, Piedras Negras, Pinotepa, Saltillo, San Luis Potosí, Tapachula, Tijuana, Torreón, Tuxtepec, Superior de Valladolid, Valle del Guadiana, Superior de Zacapoaxtla y Zacatecas.	Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de la Carrera de Ingeniería Informática.
Desarrollo de Programas en Competencias Profesionales por los Institutos Tecnológicos del 12 de octubre de 2009 al 19 de febrero de 2010.	Academias de Ingeniería Informática de los Institutos Tecnológicos de: El Llano, Colima y Los Mochis.	Elaboración del programa de estudio propuesto en la Reunión Nacional de Diseño Curricular de la Carrera de Ingeniería Informática.
Instituto Tecnológico Superior de Poza Rica del 22 al 26 de febrero de 2010.	Representantes de los Institutos Tecnológicos de: Apizaco, Cerro Azul, Chetumal, Ciudad Juárez, Ciudad Madero, Superior de Coatzacoalcos, Colima, Comitancillo, Conkal, Durango, El Llano Aguascalientes, El Salto, Superior de Fresnillo, Huejutla, Superior de Lerdo, Los Mochis, Mexicali, Morelia, Oaxaca, Superior del Occidente del Estado de Hidalgo, Ocotlán, Orizaba, Piedras Negras, Pinotepa, Saltillo, San Luis Potosí, Tapachula, Tijuana, Torreón, Tuxtepec, Superior de Valladolid, Valle del Guadiana, Superior de Zacapoaxtla y Zacatecas.	Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de la Carrera de Ingeniería Informática.

5.- OBJETIVO GENERAL DEL CURSO

Analiza y diseña soluciones de aplicaciones distribuidas utilizando los principales middlewares para el desarrollo de sistemas cliente/servidor.

6.- COMPETENCIAS PREVIAS

- Lenguaje de programación Java, Visual C o Visual Basic
- Conocimientos básicos en sistemas operativos como Windows y Linux
- Conocimiento del protocolo de red TCP/IP.

7.- TEMARIO

Unidad	Temas	Subtemas
1.	Contexto de la programación cliente/servidor	1.1. Arquitectura del modelo cliente/servidor. 1.2. 1.2 Modelos de dos y tres capas. 1.3. 1.3 Usos y aplicaciones 1.4. 1.4 Comunicación entre programas 1.5. 1.5 Modelos de computación distribuida 1.5.1. RMI 1.5.2. COM/DCOM. 1.5.3. Servicios Web. 1.5.4. Otros.
2.	Programación Cliente-Servidor de Bajo Nivel: sockets y canales	2.1. Concepto de socket. 2.2. Dominios y Tipos de sockets. 2.3. Creación/ implementación y supresión de sockets. 2.4. Desarrollo del lado del servidor con sockets. 2.5. Desarrollo del lado del cliente con sockets.
3.	RMI (REMOTE METHOD INVOCATION)	3.1. Características y Estructura de RMI 3.2. El API Java RMI. 3.3. Jerarquía de objetos RMI. 3.4. El Sistema de Nombrado Registry. 3.5. Desarrollo de Aplicaciones Distribuidas. 3.6. Paso de parámetros a través de la red. 3.7. Callbacks (Resguardos).
4.	COM/DCOM (Component Object Model/ Distributed COM)	4.1. Creación de Servidores COM. 4.2. Creación de un cliente COM. 4.3. Automatización. 4.4. ATL (Active Template Library). 4.5. DCOM (Distributed COM).
5.	Servicios web XML	5.1. Características del lenguaje 5.2. Visión general de servicios web XML 5.3. Tecnologías subyacentes. 5.3.1. SOAP 5.3.2. WSDL 5.3.3. UDDI

8.- SUGERENCIAS DIDÁCTICAS

El docente debe:

- Fomentar el uso de las tecnologías de información y comunicación.
- Dar cabida a la flexibilidad en el seguimiento del proceso formativo y propiciar la interacción entre los estudiantes.
- Reforzar la integración y continuidad de los equipos de trabajo; propiciar la realización de investigaciones de campo.
- Tomar en cuenta el conocimiento de los estudiantes como punto de partida y como posible obstáculo para la construcción de nuevos conocimientos.
- Hacer que el estudiante se ubique en la realidad al indagar sobre las experiencias tecnológicas del ambiente externo en que se desenvuelve. Ejemplos: casos relacionados con el uso de aplicaciones de red desarrolladas en el paradigma cliente/servidor.
- Relacionar los contenidos de esta asignatura con los obtenidos en las demás del plan de estudios, reforzando la importancia de tener una visión y práctica interdisciplinaria para alcanzar las metas académicas, profesionales y empresariales.
- Motivar el desarrollo de capacidades intelectuales relacionadas con la escritura, la expresión oral y la lectura de documentos técnicos. Ejemplo: Redactar cada uno de los productos que se marcan como actividades de aprendizaje.
- Orientar al estudiante en la preservación del medio ambiente, al ver que cumpla con la normatividad relativa con la minimización del impacto ambiental negativo, al momento de realizar una innovación o cambio tecnológico en alguna empresa.
- Cuando los temas lo requieran, utilizar medios audiovisuales para una mejor comprensión del estudiante.
- Integrar equipos de trabajo en donde se compartan conocimientos y experiencias académicas y laborales.
- Discutir en grupo la información generada por los equipos de trabajo
- Propiciar el uso de las diferentes fuentes de información, tanto de índole primaria como secundaria.
- Elaboración de rúbricas.

9.- SUGERENCIAS DE EVALUACIÓN

La evaluación debe ser continua y cotidiana por lo que se debe considerar el desempeño en cada una de las actividades de aprendizaje, haciendo especial énfasis en:

- Información recabada durante las consultas e investigaciones solicitadas, plasmadas en documentos escritos también llamados productos.
- Reportes escritos de los resultados u observaciones obtenidas durante las actividades realizadas en cada unidad académica, así como de las conclusiones obtenidas de dichas observaciones.
- Descripción de otras experiencias concretas que se obtendrán al participar en discusiones, exposiciones o cualquier otro medio didáctico-profesional que trate sobre la materia y que deberán realizarse durante el curso académico.
- Exámenes teórico-prácticos para comprobar la efectividad del estudiante en la resolución de casos prácticos.
- Presentación y exposición de cada actividad de aprendizaje así como de las prácticas propuestas. Algunas se evaluarán por equipo.
- Integración de las rúbricas en un portafolio de evidencias.
- Proyectos realizados empleando cada uno de los modelos estudiados de computación distribuida.

10.- UNIDADES DE APRENDIZAJE

Unidad 1: Introducción a la programación cliente/servidor

Competencia específica a desarrollar	*Actividades de Aprendizaje*
Conocer los conceptos teóricos básicos relacionados con el modelo cliente/servidor.	• Redactar un ensayo de máximo 6 cuartillas sobre el modelo cliente/servidor, especificando las principales arquitecturas lógicas, sus ventajas y desventajas.

Unidad 2: Programación Cliente-Servidor de Bajo Nivel: sockets y canales

Competencia específica a desarrollar	*Actividades de Aprendizaje*
Identificar las características de la interfaz socket. Utilizar sockets en el desarrollo de aplicaciones básicas cliente/servidor.	• Práctica de ejercicios. Desarrollar aplicaciones utilizando la interfaz socket para la comunicación entre aplicaciones en una red de computadoras: • Aplicaciones cliente. • Aplicaciones servidor.

Unidad 3: RMI (Remote Method Invocation, Invocación Remota de Métodos)

Competencia específica a desarrollar	*Actividades de Aprendizaje*
Identificar las características, ventajas y desventajas del mecanismo RMI de Java para la intercomunicación de aplicaciones mediante la invocación de métodos remotos. Desarrollar aplicaciones empleando el mecanismo RMI.	• Práctica de ejercicios. Desarrollar programas cliente-servidor utilizando ***Remote Method Invocation (RMI)*** como tecnología de base, e incluyendo serialización de objetos, control de políticas de seguridad y generación automática de resguardos.

Unidad 4: COM/DCOM (Component Object Model/ Distributed COM)

Competencia específica a desarrollar	*Actividades de Aprendizaje*
Identificar las características, ventajas y desventajas del Modelo de Objetos de Componentes Distribuidos de Microsoft. Desarrollar aplicaciones bajo DCOM.	• Práctica de ejercicios. Desarrollar aplicaciones bajo el Modelo de Objetos de Componentes Distribuidos de Microsoft (Distributed Component Object Model, DCOM), utilizando un lenguaje visual (Visual Basic o Visual C#).

Unidad 5: Servicios web XML

Competencia específica a desarrollar	*Actividades de Aprendizaje*
Comprender el funcionamiento de un servicio web. Desarrollar librerías de métodos remotos para realizar el intercambio de información estructurada en aplicaciones cliente-servidor. Publicar un servicio web para permitir su utilización por aplicaciones cliente.	• Práctica de ejercicios. Desarrollar servicios web y publicarlos en un servidor para invocarlos desde aplicaciones cliente.

11.- FUENTES DE INFORMACIÓN

1. Decker, Hirshfield. (2001). Programación con Java. 2ª. Edición. México: International Thomsom Editores.
2. Jesús Bobadilla (2003). Java a través de ejemplos. México: Alfa Omega – RAMA.
3. Kris Jamsa.(1999). Aprenda y practique Java.Ed. Oxford.
4. Wong, Paul. Java. Ed. International Thomsom Editores.
5. Márquez, Francisco M. Unix Programación Avanzada 2ª. Edición. México: Alfaomega Ra-Ma.
6. Ryan, Timothy W. Distributed Object Technology. Prentice Hall.
7. Froute , Agustín. Java 2 Manual de Usuario y Tutorial 2ª. edición. Alfaomega Rama.

12.- PRÁCTICAS PROPUESTAS

- Proyecto. Desarrolle una aplicación chat que le permita comunicarse a los usuarios de una red, tomando en cuenta las siguientes consideraciones:
    - En cada cliente de la red deberá instalarse el cliente chat
    - El servidor de chat deberá realizar lo siguiente:
        - controlar el acceso de los usuarios solicitando su identificación
        - recibir los mensajes y distribuirlos en la terminales donde se encuentren los usuarios del chat
        - controlar la no predistribución de mensajes
        - gestionar la desconexión de un cliente del chat
        - ocultar la presencia de un cliente a otros clientes del chat
    - Presente sus resultados en la rúbrica de reporte de proyecto final y exponga sus resultados en plenaria.
- Proyecto. Desarrolle un servicio web en donde se implementen métodos de acceso a datos. Posteriormente, el servicio web deberá ser publicado en un servidor a nivel intranet.
- Proyecto. Desarrolle una aplicación cliente en donde se haga el consumo de datos a través del servicio web desarrollado en el punto anterior.
- Proyecto. Desarrolle una aplicación que, mediante RMI, obtenga información del sistema de archivos de una PC remota y genere reportes.